IN THE CLAIMS:

Please amend the claims as follows:

1. (Previously Presented) A method for decoding a signal received from a dispersive channel causing intersymbol interference, said signal encoded using an MLT-3 code, said method comprising the steps of:

generating at least one trellis representing both said MLT-3 code and said dispersive channel; and

performing joint equalization and decoding of said received signal using said trellis.

2. (Original) The method of claim 1, wherein said performing step uses a reduced complexity sequence estimation technique.

3. (Cancelled).

4. (Cancelled).

5. (Cancelled).

6. (Cancelled).

7. (Original) The method of claim 1, wherein said dispersive channel is an Ethernet channel.

8. (Previously Presented) A receiver for processing a signal received from a dispersive channel, said signal encoded using an MLT-3 code, comprising:

a sequence detector that performs joint equalization and decoding of said received signal using at least one trellis representing both said MLT-3 code and said dispersive channel.

9. (Original) The receiver of claim 8, wherein said sequence detector employs a reduced complexity sequence estimator.

10. (Original) The receiver of claim 9, wherein said reduced complexity sequence estimator employs a reduced-state trellis having a reduced number of states, wherein said reduced complexity sequence estimator further comprises:

a branch metric units (BMU) that calculates branch metrics based on said received signal;

an add-compare-select unit (ACSU) that determines the best surviving paths into said reduced states;

a survivor memory unit (SMU) that stores said best surviving paths; and

a decision-feedback unit (DFU) that takes survivor symbols from said SMU to calculate ISI estimates for said reduced states, wherein said ISI estimates are used by said BMU to calculate branch metrics for transitions in the reduced-state trellis.

11. (Cancelled).

12. (Currently Amended) The receiver of claim ~~11~~ 8, wherein said sequence detector further comprises:

a branch metric units (BMU) that calculates branch metrics based on said received signal;

an add-compare-select unit (ACSU) that determines the best surviving paths into said trellis states; and

a survivor memory unit (SMU) that stores said best surviving paths.

13. (Cancelled).

14. (Cancelled).

15. (Original) The receiver of claim 8, wherein said dispersive channel is an Ethernet channel.

16. (Currently Amended) A method for representing an MLT-3 code as a trellis, said MLT-3 code using three signal levels to represent two binary values, said method comprising the steps of:

generating said trellis with a plurality of trellis states, each of said trellis states associated with a value for a signal in a previous symbol period; and

generating each of said trellis states with at least two branches leaving or entering each state, each of said at least two branches corresponding to state transitions associated with said two binary values, wherein a first binary value <u>substantially always</u> causes a state transition in said trellis <u>from a first state to a different state</u> and a second binary value does not cause a state transition in said trellis.

17. (Original) The method of claim 16, wherein a first one of said plurality of trellis states corresponds to a value for a signal in a previous symbol period of +1.

18. (Original) The method of claim 16, wherein a second and third of said plurality of trellis states corresponds to a value for a signal in a previous symbol period of 0.

19. (Original) The method of claim 16, wherein a fourth one of said plurality of trellis states corresponds to a value for a signal in a previous symbol period of -1.

20. (Original) The method of claim 16, further comprising the step of using said trellis to perform joint equalization and decoding of a signal encoded using said MLT-3 code.

21. (Previously Presented) The method of claim 16, further comprising the step of combining said trellis with a trellis representing a channel to obtain a super trellis.

22. (Original) The method of claim 16, wherein said dispersive channel is an Ethernet channel.

23. (Previously Presented) The method of claim 1, wherein a state in said trellis is given by a concatenation of said MLT-3 code state and a channel state, wherein said channel state describes said dispersive channel.

24. (Previously Presented) The method of claim 1, wherein a state in said trellis is given by a concatenation of said MLT-3 code state and a truncated channel state, wherein said truncated channel state partially describes said dispersive channel.

25. (Previously Presented) The method of claim 24, further comprising the steps of computing ISI estimates for said states using symbols from corresponding survivor paths; computing branch metrics for transitions in said trellis based on said ISI estimates; determining survivor paths into said states based on said branch metrics; and storing said survivor paths.

26. (Previously Presented) The method of claim 24, wherein a number of states in said trellis is given by $4x(2^K)$, where K is the truncated channel memory.

27. (Previously Presented) The receiver of claim 8, wherein a state in said trellis is given by a concatenation of said MLT-3 code state and a channel state, wherein said channel state describes said dispersive channel.

28. (Previously Presented) The receiver of claim 8, wherein a state in said trellis is given by a concatenation of said MLT-3 code state and a truncated channel state, wherein said truncated channel state partially describes said dispersive channel.

29. (Previously Presented) The receiver of claim 28, wherein a number of states in said trellis is given by $4x(2^K)$, where K is the truncated channel memory.